| | | |
|---|---|---|
| First Named Inventor | James A. Lamb | **APPEAL BRIEF** |
| Serial No. | 10/692,939 | |
| Filing Date | October 24, 2003 | |
| Group Art Unit | 2194 | |
| Examiner Name | Charles E. Anya | |
| Confirmation No. | 2113 | |
| Attorney Docket No. | 200312021-1 | |
| Title: PROGRAM INTERFACE ARCHITECTURE | | |

# APPEAL BRIEF

Mail Stop: Appeal Brief - Patents
Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

## Table of Contents

## I. Introduction

In response to the Final Office Action mailed January 12, 2009, Appellant filed a Notice of Appeal to the Board of Patent Appeals and Interferences on February 12, 2009. A request for extension of time and accompanying fee has previously been submitted, thereby extending the period for submission of this Appeal Brief to August 12, 2009. One copy of this Appeal Brief is hereby filed, in accordance with 37 C.F.R. § 41.37(a)(1), and is accompanied by an authorization to charge Appellant's deposit account for the fee in the amount of $540.00 as required under 37 C.F.R. § 41.20(b)(2).

## II. Real Party in Interest

The real party in interest in the above-captioned application is the assignee Hewlett-Packard Development Company, L.P., a Texas Limited Partnership having its principal place of business at 20555 SH 249, Houston, TX 77070, in an assignment recorded on October 24, 2003 at Reel 014647, Frame 0318.

## III. Related Appeals and Interferences

There are no other appeals or interferences known to Appellant that will have a bearing on the Board's decision in the present Appeal.

## IV. Status of Claims

Claims 1-11 and 14-32 are pending in the application. Claims 12-13 were previously canceled. Claims 1-11 and 14-32 are the subject of this Appeal.

In the Final Office Action mailed January 7, 2009, claims 17-28 were objected to because of informalities; claims 17-23 and 29-32 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention; claims 1-3, 5-7, 9 and 23-27 were rejected under 35 U.S.C. § 102(e) as being anticipated by Bond et al. (U.S. Publication No. 2002/0052727A1); claim 4 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Gessel et al. (U.S. Patent No. 5,889,954); claims 8 and 28 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Schuetz et al. (U.S. Patent No. 6,725,451); claims 10, 11, 14-18, 20, 22 and 29-32 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S Statutory Invention Registration No. H1,921); claims 19 and 21 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S Statutory Invention Registration No. H1,921) as applied to claim 17 above, and further in view of Schuetz et al. (U.S. Patent No. 6,725,451). No claims have been indicated as being allowable.

See Appendix A for claims 1-11 and 14-32 involved in this Appeal.

## V. Status of Amendments

All amendments to the claims have been entered.

## VI. Summary of Claimed Subject Matter

The present claims are directed to methods and apparatus using an interface module 118 to enable a program application in the application layer 116 to properly execute with one type of operating system in the operating system layer 120 which is different from an operating system for which the program application was written. Specification, page 5, lines 11-14 and Fig. 1

The subject matter defined in claim 1 involved in this appeal includes a computing device 110 comprising a processor and memory having instructions stored therein, that are executable by the processor. Specification, page 1, lines 4-6 and page 20, lines 15-20. The computing device 110 provides an application layer 116 having executable instructions to provide one or more applications and an operating system layer 120 having executable instructions to provide a first type of operating system and associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device. Specification, page 4, line 27 through page 5, line 1; and Fig. 1. The computing device 110 also includes an interface module 118 coupled between the application layer 116 and the operating system layer 120, wherein the interface module 118 includes a second number of APIs for operating with a second type of operating system that is not the type of operating system resident on the computing device 110 and wherein the interface module 118 receives program instructions from a program in the application layer 116 written for a second type of operating system. Specification, page 5, lines 8-14; page 6, lines 28-30; page 9, lines 25-32; and Figs. 1-3. The interface module 118 processes the instructions to select either, one of the first number of APIs or one of the second number of APIs. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; and page 14, lines 14-16.

The subject matter defined in claim 10 involved in this appeal includes a system architecture comprising a computing device 110 including a processor and memory having instructions stored therein, that are executable by the processor. Specification, page 1, lines 4-6 and page 20, lines 15-20. The computing devices 110 provides an application layer 116 having a home location register application thereon and having executable instructions to provide one or more applications. Specification, page 4, lines 25-31; and Fig. 1. The system architecture includes an operating system layer 120 having executable instructions to provide a first type of operating system and associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device 110. Specification, page 4, lines 27 through page 5, line 1; and Fig. 1. The computing device 110 includes an interface module 118 coupled between the application layer 116 and the operating system layer 120, wherein the interface module 118 includes a second number of APIs for operating the home location register application with a second type of operating system that is not the type of operating system resident on the computing device 110 and wherein the interface module 118 receives program instructions from the home location register application in the application layer 116 written for the second type of operating system and processes the instructions to select either one of the first number of APIs or one of the second number of APIs. Specification, page 4, line 32 through page 5, line 14; page 6, lines 28-30; page 7, lines 6-8; page 9, lines 25-32; page 11, lines 14-20; and Figs. 1-3. The interface module 118 also includes a connection for connecting the computing device 110 to a publicly switched telephone network (PSTN). Specification, page 18, lines 25-28.

Independent claim 17 recites a method of executing an application comprising providing an application via an application layer 116 having executable instructions to provide one or more applications to an operating system layer 120 having executable instructions to provide a first type of operating system and associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device 110. Specification, page 4, lines 27 through page 5, line 1; and Fig. 1. The

method further includes communicating instructions stored in memory and executable on a processor from the application to an interface module 118. Specification, page 1, lines 14-17; and page 4, lines 28-31. The interface module 118 includes a second number of APIs for operating the home location register application with a second type of operating system that is not the type of operating system resident on the computing device 110. Specification, page 5, lines 8-14; page 6, lines 28-30; page 9, lines 25-32; and Figs. 1-3. The interface module 118 receives program instructions from the application in the application layer written for the second type of operating system and processes the instructions to select either, one of the first number of APIs or one of the second number of APIs. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; and page 14, lines 14-16.

Independent claim 23 recites a method of executing an application comprising communicating instructions from the application to an interface module 118. Specification, page 5, lines 28-33; and Fig. 1. The application is configured for a first type of operating system. Specification, page 5, lines 28-33. The method also includes interpreting the instructions from the application with the interface module 118 by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on the second type of operating system and wherein the second number of APIs are resident on the interface module 118. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; and page 14, lines 14-16. The method further includes communicating the instructions from the interface module 118 to an operating system that is the second type of operating system. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; page 14, lines 14-16; and Fig. 1.

Independent claim 29 recites a computer readable medium having a set of computer executable instructions thereon for causing a device 110 to perform a method, comprising communicating instructions from a telecommunications application to an interface module 118, the telecommunication application configured for a first type of operating system. Specification, page 4, lines 25-32; and Fig. 1. The method performed by the device also includes processing the instructions from the telecommunication application with the interface module by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on a second type of operating system and wherein the second number of APIs are resident on the interface module 118. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; and page 14, lines 14-16. The method further includes communicating the instructions from the interface module 118 to an operating system that is the second type of operating system. Specification, page 5, lines 1-4; page 7, lines 7-8 and 21-25; page 8, lines 2-11 and 29-30; page 14, lines 14-16; and Fig. 1.

## VII. Grounds of Rejection to be Reviewed on Appeal

- Whether claims 17-28 were properly objected to because of informalities.
- Whether claims 17-23 and 29-32 were properly rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.
- Whether claims 1-3, 5-7, 9 and 23-27 were properly rejected under 35 U.S.C. § 102(e) as being anticipated by Bond et al. (U.S. Publication No. 2002/0052727A1).
- Whether claim 4 was properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Gessel et al. (U.S. Patent No. 5,889,954).
- Whether claims 8 and 28 were properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Schuetz et al. (U.S. Patent No. 6,725,451).
- Whether claims 10, 11, 14-18, 20, 22 and 29-32 were properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S. Statutory Invention Registration No. H1,921).
- Whether claims 19 and 21 were properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S. Statutory Invention Registration No. H1,921) as applied to claim 17 above, and further in view of Schuetz et al. (U.S. Patent No. 6,725,451).

## VIII. Argument

### A. Applicable Authorities

35 U.S.C. § 102

A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California,* 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987); (*See,* MPEP §2131). "The identical invention must be shown in as complete detail as is contained in the . . . claim." *Richardson v. Suzuki Motor Co.,* 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989); (*See,* MPEP §2131). The elements must be arranged as required by the claim, but identical terminology is not required. *In re Bond*, 910 F. 2d 831, 15 U.S.P.Q.2d 1566 (Fed. Cir. 1990); (*See,* MPEP §2131).

Anticipation focuses on whether a claim reads on a product or process disclosed in a prior art reference, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.,* 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). To anticipate a claim, a reference must disclose every element of the challenged claim and enable one skilled in the art to make the anticipating subject matter. *PPG Industries, Inc. v. Guardian Industries Corp.,* 75 F.3d 1558, 37 U.S.P.Q.2d 1618 (Fed. Cir. 1996).

"For a prior art reference to anticipate a claim, the reference must disclose each and every element of the claim with sufficient clarity to prove its existence in the prior art. . . . Although this disclosure requirement presupposes the knowledge of one skilled in the art of the claimed invention, that presumed knowledge does not grant a license to read

into the prior art reference teachings that are not there." *Motorola, Inc. v. Interdigital Tech. Corp.*, 121 F.3d 1461, 43 USPQ 2d 1481, 1490 (Fed. Cir. 1997).

35 U.S.C. § 103

35 U.S.C.§103(a) provides in relevant part:

> Conditions for patentability; non-obvious subject matter.
>
> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

"The ultimate determination … whether an invention is or is not obvious is a legal conclusion based on underlying factual inquiries including: (1) the scope and content of the prior art; (2) the level of ordinary skill in the prior art; (3) the differences between the claimed invention and the prior art; and (4) objective evidence of nonobviousness." *In re Dembiczak*, 175 F.3d 994, 998, 50 USPQ2d 1614, 1616 (Fed. Cir. 1999) (citing *Graham v. John Deere Co.*, 383 U.S. 1, 17-18, 148 USPQ 459, 467 (1966)); *see also KSR International Co. v. Teleflex Inc.*, 127 S.Ct. 1727, 1734, 82 USPQ2d 1385, 1391 (2007) (reaffirming that the *Graham* factors continue to control the determination of obviousness).

When applying 35 U.S.C. §103, the claimed invention must be considered as a whole; the references must be considered as a whole and must suggest the desirability and thus the obviousness of making the combination; the references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention and a reasonable expectation of success is the standard with which obviousness is

determined. *Hodosh v. Block Drug Co., Inc.*, 786 F.2d 1136, 1143 n.5, 229 USPQ 182, 187 n.5 (Fed. Cir. 1986).

To establish a *prima facie* case of obviousness, three basic criteria must be met: (1) There must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings; (2) There must be a reasonable expectation of success; (3) The prior art references must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on appellants' disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). *But see*, *KSR International Co. v. Teleflex Inc.*, 127 S.Ct. 1727, 1734, 82 USPQ2d 1385, 1391 (2007) (cautioning against the rigid application of a teaching/suggestion/motivation rule).

If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. *See, e.g., In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959) (Claims were directed to an oil seal comprising a bore engaging portion with outwardly biased resilient spring fingers inserted in a resilient sealing member. The primary reference relied upon in a rejection based on a combination of references disclosed an oil seal wherein the bore engaging portion was reinforced by a cylindrical sheet metal casing. Patentee taught the device required rigidity for operation, whereas the claimed invention required resiliency. The court reversed the rejection holding the "suggested combination of references would require a substantial reconstruction and redesign of the elements shown in [the primary reference] as well as a change in the basic principle under which the [primary reference] construction was designed to operate." *Id.*, 270 F.2d at 813, 123 USPQ at 352.).

MPEP § 707.07(f)

A preamble is generally not accorded any patentable weight . . . where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone. *See In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

MPEP § 2173.05(e)

A claim is indefinite when it contains words or phrases whose meaning is unclear. The lack of clarity could arise where a claim refers to "said lever" or "the lever," where the claim contains no earlier recitation or limitation of a lever and where it would be unclear as to what element the limitation was making reference. Similarly, if two different levers are recited earlier in the claim, the recitation of "said lever" in the same or subsequent claim would be unclear where it is uncertain which of the two levers was intended. A claim which refers to "said aluminum lever," but recites only "a lever" earlier in the claim, is indefinite because it is uncertain as to the lever to which reference is made. Obviously, however, the failure to provide explicit antecedent basis for terms does not always render a claim indefinite. If the scope of a claim would be reasonably ascertainable by those skilled in the art, then the claim is not indefinite. *Energizer Holdings Inc. v. Int'l Trade Comm'n*, 435 F.3d 1366, 77 USPQ2d 1625 (Fed. Cir. 2006) (holding that "anode gel" provided by implication the antecedent basis for "zinc anode"); *Ex parte Porter*, 25 USPQ2d 1144, 1145 (Bd. Pat. App. & Inter. 1992) ("controlled stream of fluid" provided reasonable antecedent basis for "the controlled fluid"). Inherent components of elements recited have antecedent basis in the recitation of the components themselves. For example, the limitation "the outer surface of said sphere" would not require an antecedent recitation that the sphere has an outer surface. *See Bose Corp. v. JBL, Inc.*, 274 F.3d 1354, 1359, 61 USPQ2d 1216, 1218-19 (Fed. Cir 2001) (holding that recitation of "an ellipse" provided antecedent basis for "an ellipse having a major

diameter" because "[t]here can be no dispute that mathematically an inherent characteristic of an ellipse is a major diameter").

Support for MPEP as Authority

The MPEP is "commonly relied upon as a guide to patent attorneys and patent examiners on procedural matters." *Mollins PLC v. Textron, Inc.*, 48 F.3d 1172, 1180 n.10 (Fed. Cir. 1995) (internal quotation marks omitted). The MPEP "does not have the force of law, [but] it is entitled to judicial notice as an official interpretation of statutes or regulations as long as it is not in conflict therewith." *Id.* The Federal Circuit has noted that the "PTO operates in accordance with detailed rules and regulations, including those set out in the [MPEP] which is made available to the public and which has been held to describe procedures on which the public can rely." *Patlex Corp. v. Mossinghoff*, 758 F.2d 594, 606 (Fed. Cir. 1985).

Allowability of Dependent Claims

If an independent claim is non-obvious under 35 U.S.C. § 103, then any claim depending therefrom is non-obvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

**B. Analysis**

1. Claim Objections

a) Claims 17-22

Independent claim 17 was objected to, the Final Office Action asserting a typographical error. Specifically, the Final Office Action stated that is was unclear whether "an application" is different from "one or more applications" and whether a relationship existed between the "application layer" and the "first type of operating system." Final Office Action, page 2, section 2, first paragraph. Appellant contends that it is clear from the context of the claim, and the use within Appellant's Specification, that the application layer has executable instructions to provide one or more applications to an operating system layer, and that the application being provided by the application layer is one of those one or more applications. Appellant further contends that it is clear from the context of the claim, and the use within Appellant's Specification, that the operating system layer has executable instructions to provide a first type of operating system and associated application program interfaces (APIs), and that, aside from being in communication with and responding to the application layer, there is no recited relationship between the application layer and the first type of operating system. There is no explicit objection of claims 18-22, and Appellant presumes that the only objection is through their dependence on claim 17. Appellant thus respectfully requests that the Board of Appeals remove the Examiner's objection regarding claims 17-22.

b) Claim 23-28

Independent claim 23 was objected to as containing a typographical error. Specifically, the Final Office Action noted that the article "a" should have preceded "first operating system" in line 2 of claim 23. *See* Final Office Action, page 3, first paragraph. Appellant acknowledges this omission, but believes the error is *de minimis* and contends

that the error results in no ambiguity or indefiniteness in the claim as presented. Appellant contends that the body of claim 23 does not depend on its preamble for completeness but, instead, the process steps are able to stand alone. As such, the missing article does not affect the scope of claim 23. *See* MPEP § 707.07(f) (citing *In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951)). Appellant further authorizes an Examiner's Amendment to amend claim 23 as proposed in the Final Office Action if necessary to maintain the appeal of claim 23 on the merits. *See* Final Office Action, page 3, second paragraph. There is no explicit objection of claims 24-28, and Appellant presumes that the only objection is through their dependence on claim 23. Appellant thus respectfully requests that the Board of Appeals remove the Examiner's objection regarding claims 23-28.

2. Claim Rejections Under 35 U.S.C. § 112

Independent claims 17, 23 and 29, and dependent claims 18-22 and 30-32, were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Dependent claims 18-22 and 30-32 are not separately argued under the provisions of 37 CFR 41.37(c)(1)(vii).

a) Claims 17-22

The Final Office Action noted a lack of antecedent basis for "the computing device" on line 8 (line 7 in Appendix A) of claim 17. Final Office Action, page 3, section 3(i). The Final Office Action notes that the Examiner would interpret and replace "the computing device" with "a computing device." *Id.* The Final Office Action further noted a lack of antecedent basis for "the home location register application" on line 11 (line 10 in Appendix A) of claim 17. Final Office Action, page 3, section 3(ii). The Final Office Action notes that the Examiner would interpret and replace "the home location register application" with "a home location register application." *Id.* Appellant requests that the Board accept the Examiner's interpretation for purposes of this appeal as

Appellant contends that the scope of the claim is reasonably ascertainable as evidenced by the Examiner's interpretation. *See* MPEP § 2173.05(e) ("If the scope of a claim would be reasonably ascertainable by those skilled in the art, then the claim is not indefinite."). Appellant further authorizes an Examiner's Amendment to amend claim 17 as proposed in the Final Office Action if necessary to maintain the appeal of claim 17 with respect to the rejection under 35 U.S.C. § 103(a). There is no explicit rejection of claims 18-22 under 35 U.S.C. § 112, second paragraph, and Appellant presumes that the only rejection under 35 U.S.C. § 112, second paragraph is through their dependence on claim 17. Appellant thus respectfully requests that the Board of Appeals remove the rejection under 35 U.S.C. § 112, second paragraph of claims 17-22.

b) Claim 23

The Final Office Action provides no basis for rejection of claim 23 under 35 U.S.C. § 112, second paragraph. As such, Appellant requests that the Board remove the rejection of claim 23 under 35 U.S.C. § 112, second paragraph as unsupported by the record.

c) Claim 29-32

The Final Office Action noted a lack of antecedent basis for "the application" on line 8 (line 7 in Appendix A) of claim 29. Final Office Action, page 3, section 3(iii). The Final Office Action notes that the Examiner would interpret and replace "the application" with "the telecommunication application." *Id.*

Appellant contends that there is no lack of antecedent basis as a telecommunication application is inherently an application. As there is no recitation of any other application, Appellant contends that it is clear that "the application" occurring in line 7 of claim 29 in Appendix A refers only to the telecommunications application recited earlier in that element of claim 29. As such, Appellant contends that claim 29 is not indefinite. *See* MPEP § 2173.05(e) ("If the scope of a claim would be reasonably ascertainable by those skilled in the art, then the claim is not indefinite."). There is no

explicit rejection of claims 30-32 under 35 U.S.C. § 112, second paragraph, and Appellant presumes that the only rejection under 35 U.S.C. § 112, second paragraph is through their dependence on claim 29. Appellant thus respectfully requests that the Board of Appeals remove the rejection under 35 U.S.C. § 112, second paragraph of claims 29-32. Appellant further authorizes an Examiner's Amendment to amend claim 29 as proposed in the Final Office Action if necessary to maintain the appeal of claim 29 with respect to the rejection under 35 U.S.C. § 103(a).

3. Claim Rejections Under 35 U.S.C. § 102

Independent claims 1 and 23, and dependent claims 2-3, 5-7, 9 and 24-27, were rejected under 35 U.S.C. § 102(e) as being anticipated by Bond et al. (U.S. Publication No. 2002/0052727A1). Appellant contends that the rejection of the independent claims 1 and 23 is unsupported by the record as discussed below. Dependent claims 2-3, 5-7, 9 and 24-27 are not separately argued under the provisions of 37 CFR 41.37(c)(1)(vii).

a) Claims 1-3, 5-7 and 9

The Final Office Action identifies the non-native kernel emulator 400 of Bond et al. as corresponding to Appellant's interface module in claim 1. Final Office Action, page 5, first full paragraph. However, claim 1 recites in part, "wherein the interface module includes a second number of APIs for operating with a second type of operating system that is not the type of operating system resident on the computing device and wherein the interface module receives program instructions from a program in the application layer written for the second type of operating system and processes the instructions to select either, one of the first number of APIs or one of the second number of APIs." Appellant first notes that the non-native kernel emulator 400 of Bond et al. does not include any APIs, but instead receives and acts upon non-native APIs of Bond et al. *See, e.g.,* Bond et al., paragraph 0071. Thus, the non-native kernel emulator 400 cannot correspond to Appellant's interface module as it fails to include the second number of APIs for operating with a second type of operating system as recited in

Appellant's claim 1. In addition, neither the non-native kernel emulator 400 nor any other element or combination of elements of Bond et al. receives program instructions and processes those instructions to select either an API from a first number of APIs or an API from a second number of APIs. Instead, Bond et al. expressly teaches that native applications exclusively call the set of native APIs 320 and that it modifies stub code of the non-native APIs 322 of non-native applications to specifically call its non-native kernel emulator 400. *See* Bond et al., paragraphs 0069-0071. Thus, the cited reference does not teach receiving instructions from a program and selecting an API from two sets of APIs, but teaches that the program calls a first set of APIs 320 if that program is written for the operating system of the computing device and that the program calls a second set of APIs 322 if that program is written for a different operating system. In addition, this decision is made prior to executing program instructions of the application, and thus cannot be said to be processing the instructions to select any APIs. *See* Bond et al., Figure 5A and paragraphs 0129-0131 (noting that the decision to operate as a native or non-native application occurs prior to running the application). As such, Appellant contends that there is no element or combination of elements of Bond et al. that receives and processes program instructions to select either one of a first number of APIs or one of a second number of APIs as recited in Appellant's claim 1. In view of the foregoing, Appellant contends that the rejection under 35 U.S.C. § 102(e) must fail as the cited reference fails to teach each and every element of Appellant's claim 1.

Although not asserted by the Final Office Action, Appellant further contends that Bond et al. would fail to render obvious the elements of Appellant's claim 1 as it would necessarily change a principle of operation of Bond et al. if the reference were modified in a manner necessary to support a rejection under 35 U.S.C. § 103. In particular, Bond et al. processes instructions from its applications as either native or non-native applications and calls native or non-native APIs, respectively, based on the decision whether the application is native or non-native. In contrast, a modification necessary to support a rejection under 35 U.S.C. § 103 would require Bond et al. to process instructions from both native and non-native applications in a like manner, and make its decision on which API to use based on the processed instructions rather than the native / non-native application determination made during initialization of an application. Thus,

Appellant contends that Bond et al. could not be modified in a manner necessary to support a rejection under 35 U.S.C. § 103. *See, e.g., In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959).

Because Bond et al. fails to teach each and every element of Appellant's claim 1, and because a modification of Bond et al. is impermissible, Appellant contends that claim 1 is allowable over Bond et al. Furthermore, if an independent claim is allowable, then any claim depending therefrom is allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Thus, for all the reasons stated above with reference to claim 1, Appellant contends that remaining dependent claims 2, 3, 5-7 and 9 are also allowable. In view of the foregoing, Appellant thus respectfully requests reversal of the rejection of claims 1-3, 5-7 and 9.

b) Claims 23-27

The Final Office Action identifies the non-native kernel emulator 400 of Bond et al. as corresponding to Appellant's interface module in claim 23. Final Office Action, page 7, section 11, second paragraph. However, claim 23 recites in part, "interpreting the instructions from the application with the interface module by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on the second type of operating system and wherein the second number of APIs are resident on the interface module." As noted with reference to claim 1, the non-native kernel emulator 400 of Bond et al. fails to teach or render obvious receiving program instructions from an application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system or one of a second number of APIs that are designed for use of the application on the second type of operating system.

In view of the foregoing, Appellant contends that claim 23 is allowable over the Bond et al. reference. Furthermore, if an independent claim is allowable, then any claim

depending therefrom is allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Thus, for all the reasons stated above with reference to claim 23, Appellant contends that remaining dependent claims 24-27 are also allowable. Appellant thus respectfully requests reversal of the rejection of claims 23-27.

4. Claim Rejections Under 35 U.S.C. § 103

a) Claim 4

Dependent claim 4 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Gessel et al. (U.S. Patent No. 5,889,954).

Appellant contends that it has shown claim 1 to be patentably distinct from the primary reference of Bond et al. In particular, Bond et al. fails to teach or render obvious an interface module including a second number of APIs for operating with a second type of operating system that is not the type of operating system resident on a computing device, wherein the interface module receives program instructions from a program in an application layer written for the second type of operating system and processes the instructions to select either, one of a first number of APIs or one of a second number of APIs as recited in claim 1. As noted with reference to claim 1, Appellant contends that modification of Bond et al. is improper. However, even if modification were proper, the secondary reference of Gessel et al. is not purported to cure the deficiencies of the primary reference with respect to claim 1, and Appellant contends that it cannot do so. As such, Appellant contends that claim 1 remains patentably distinct from Bond et al. in view of Gessel et al. As claim 4 includes all patentable elements of claim 1, this claim is also allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Appellant thus respectfully requests reversal of the rejection of claim 4.

b) Claims 8 and 28

Dependent claims 8 and 28 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Schuetz et al. (U.S. Patent No. 6,725,451).

Appellant contends that it has shown claims 1 and 23 to be patentably distinct from the primary reference of Bond et al. In particular, Bond et al. fails to teach or render obvious an interface module including a second number of APIs for operating with a second type of operating system that is not the type of operating system resident on a computing device, wherein the interface module receives program instructions from a program in an application layer written for the second type of operating system and processes the instructions to select either, one of a first number of APIs or one of a second number of APIs as recited in claim 1. In addition, Bond et al. fails to teach or render obvious interpreting instructions from an application with an interface module by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on a first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on a second type of operating system and wherein the second number of APIs are resident on the interface module as recited in claim 23. As noted with reference to claim 1, Appellant contends that modification of Bond et al. is improper. However, even if modification were proper, the secondary reference of Schuetz et al. is not purported to cure the deficiencies of the primary reference with respect to claims 1 and 23, and Appellant contends that it cannot do so. As such, Appellant contends that claims 1 and 23 remain patentably distinct from Bond et al. in view of Schuetz et al. As claim 8 includes all patentable elements of claim 1, and claim 28 includes all patentable elements of claim 23, these claims are also allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Appellant thus respectfully requests reversal of the rejection of claims 8 and 28.

c) Claims 10, 11, 14-18, 20, 22 and 29-32

Independent claims 10, 17 and 29, and dependent claims 11, 14-16, 18, 20, 22 and 30-32 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S Statutory Invention Registration No. H1,921). Appellant contends that the rejection of the independent claims 10, 17 and 29 is unsupported by the record as discussed below. Dependent claims 11, 14-16, 18, 20, 22 and 30-32 are not separately argued under the provisions of 37 CFR 41.37(c)(1)(vii).

i) Claims 10, 11 and 14-16

The Final Office Action identifies the non-native kernel emulator 400 of Bond et al. as corresponding to Appellant's interface module in claim 10. Final Office Action, page 12, first paragraph. However, claim 10 recites in part, "wherein the interface module includes a second number of APIs for operating the home location register application with a second type of operating system that is not the type of operating system resident on the computing device and wherein the interface module receives program instructions from the home location register application in the application layer written for the second type of operating system and processes the instructions to select either one of the first number of APIs or one of the second number of APIs." As noted with reference to claim 1, the non-native kernel emulator 400 of Bond et al. fails to teach or render obvious receiving program instructions from an application and processing the instructions to select either one of a first number of APIs or one of a second number of APIs. Also as noted with reference to claim 1, modification of Bond et al. would impermissibly change a principle of operation of this primary reference. Furthermore, even if a combination of Bond et al. and Fletcher et al. were permissible, which Appellant denies, the combination would still fail to teach or render obvious the elements of claim 10 as Fletcher et al. is not purported to overcome the deficiencies of the Bond et al. reference with respect to at least these elements of claim 10, and Appellant contends that it cannot do so. In particular, the combination of Bond et al. and Fletcher et al. does not

teach or render obvious receiving program instructions from an application and processing the instructions to select either one of a first number of APIs or one of a second number of APIs as recited in Appellant's claim 10.

In view of the foregoing, Appellant contends that claim 10 is allowable over Bond et al. and Fletcher et al., taken either alone or in combination. Furthermore, if an independent claim is allowable, then any claim depending therefrom is allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Thus, for all the reasons stated above with reference to claim 10, Appellant contends that remaining dependent claims 11 and 14-16 are also allowable. Appellant thus respectfully requests reversal of the rejection of claims 10, 11 and 14-16.

ii) Claims 17, 18, 20 and 22

The Final Office Action identifies the non-native kernel emulator 400 of Bond et al. as corresponding to Appellant's interface module in claim 17. Final Office Action, page 14, section 27. However, claim 17 recites in part, "wherein the interface module receives program instructions from the application in the application layer written for the second type of operating system and processes the instructions to select either, one of the first number of APIs or one of the second number of APIs." As noted with reference to claim 1, the non-native kernel emulator 400 of Bond et al. fails to teach or render obvious receiving program instructions from an application and processing the instructions to select either one of a first number of APIs or one of a second number of APIs. Also as noted with reference to claim 1, modification of Bond et al. would impermissibly change a principle of operation of this primary reference. Furthermore, even if a combination of Bond et al. and Fletcher et al. were permissible, which Appellant denies, the combination would still fail to teach or render obvious the elements of claim 17 as Fletcher et al. is not purported to overcome the deficiencies of the Bond et al. reference with respect to at least these elements of claim 17, and Appellant contends that it cannot do so. In particular, the combination of Bond et al. and Fletcher et al. does not teach or render obvious receiving program instructions from an application in an application layer written for a second type

of operating system and processing the instructions to select either, one of a first number of APIs or one of a second number of APIs as recited in Appellant's claim 17.

In view of the foregoing, Appellant contends that claim 17 is allowable over Bond et al. and Fletcher et al., taken either alone or in combination. Furthermore, if an independent claim is allowable, then any claim depending therefrom is allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Thus, for all the reasons stated above with reference to claim 17, Appellant contends that remaining dependent claims 18, 20 and 22 are also allowable. Appellant thus respectfully requests reversal of the rejection of claims 17, 18, 20 and 22.

iii) Claims 29-32

Claim 29 recites, in part, "receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on a second type of operating system and wherein the second number of APIs are resident on the interface module." Appellant contends that it has shown, with reference to claim 1, that the primary reference of Bond et al. fails to teach or render obvious at least these elements of Appellant's claim 29. Also as noted with reference to claim 1, modification of Bond et al. would impermissibly change a principle of operation of this primary reference. Furthermore, even if a combination of Bond et al. and Fletcher et al. were permissible, which Appellant denies, the combination would still fail to teach or render obvious the elements of claim 29 as Fletcher et al. is not purported to overcome the deficiencies of the Bond et al. reference with respect to at least these elements of claim 29, and Appellant contends that it cannot do so. In particular, the combination of Bond et al. and Fletcher et al. does not teach or render obvious receiving program instructions from an application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on a first type of operating system or one of a second number of APIs that are

designed for use of the application on a second type of operating system as recited in Appellant's claim 29.

In view of the foregoing, Appellant contends that claim 29 is allowable over Bond et al. and Fletcher et al., taken either alone or in combination. Furthermore, if an independent claim is allowable, then any claim depending therefrom is allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Thus, for all the reasons stated above with reference to claim 29, Appellant contends that remaining dependent claims 30-32 are also allowable. Appellant thus respectfully requests reversal of the rejection of claims 29-32.

d) Claims 19 and 21

Dependent claims 19 and 21 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Bond et al. (U.S. Publication No. 2002/0052727A1) in view of Fletcher et al. (U.S Statutory Invention Registration No. H1,921) as applied to claim 17 above, and further in view of Schuetz et al. (U.S. Patent No. 6,725,451).

Appellant contends that it has shown claim 17 to be patentably distinct from the primary reference of Bond et al. and the secondary reference of Fletcher et al., taken either alone or in combination. In particular, Bond et al. and Fletcher et al. fail to teach or render obvious receiving program instructions from an application and processing the instructions to select either, one of a first number of APIs or one of a second number of APIs as recited in claim 17. The tertiary reference of Schuetz et al. is not purported to cure the deficiencies of the primary and secondary references with respect to claim 17, and Appellant contends that it cannot do so. As such, Appellant contends that claim 17 remains patentably distinct from Bond et al. in view of Fletcher et al. and further in view of Schuetz et al. As claims 19 and 21 include all patentable elements of claim 17, these claims are also allowable. *See In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Appellant thus respectfully requests reversal of the rejection of claims 19 and 21.

**IX. Conclusion**

Appellant contends that it has shown the cited references to be deficient and unable to teach or render obvious each and every element of Appellant's claims 1-11 and 14-32.

For at least the reasons discussed above, Appellant submits that the pending claims are patentable. Accordingly, Appellant requests that the Board of Appeals reverse the Examiner's decisions regarding claims 1-11 and 14-32.

Respectfully submitted,

Date: 12/4/09

Thomas W. Leffert

Reg. No. 40,697

Attorneys for Appellant
HEWLETT-PACKARD COMPANY
Intellectual Property Administration
3404 East Harmony Rd.
Fort Collins, CO 80527-2400

## APPENDIX A
## Claims Appendix

1. A computing device, comprising:

a processor; and

memory having instructions stored therein, that are executable by the processor, to provide:

an application layer having executable instructions to provide one or more applications;

an operating system layer having executable instructions to provide a first type of operating system and associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device; and

an interface module coupled between the application layer and the operating system layer, wherein the interface module includes a second number of APIs for operating with a second type of operating system that is not the type of operating system resident on the computing device and wherein the interface module receives program instructions from a program in the application layer written for the second type of operating system and processes the instructions to select either, one of the first number of APIs or one of the second number of APIs.

2. The computing device of claim 1, wherein the interface module includes an operating system emulation module for emulating a number of operating system functions.

3. The computing device of claim 1, wherein the interface module emulates operating system functions and network server functions.

4. The computing device of claim 1, wherein the interface module emulates home location register functions.

5. The computing device of claim 1, wherein the interface module emulates intelligent network server functions.

6. The computing device of claim 1, wherein the interface module has portions for emulating the operating system functions and the network server functions in discrete modules located within the interface module.

7. The computing device of claim 1, wherein the interface module processes a program instruction by interpreting whether the instruction has to be processed further.

8. The computing device of claim 7, wherein the interface module converts a result received from the operating system layer such that the converted result is in a format that the application program can use to execute the instruction.

9. The computing device of claim 7, wherein the application interface module translates the instruction received such that the operating system layer can execute the instruction.

10. A system architecture, comprising:

a computing device including:

a processor; and

memory having instructions stored therein, that are executable by the processor, to provide:

an application layer having a home location register application thereon and having executable instructions to provide one or more applications;

an operating system layer having executable instructions to provide a first type of operating system and associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device; and

an interface module coupled between the application layer and the operating system layer, wherein the interface module includes a second number of APIs for operating the home location register application with a second type of operating system that is not the type of operating system resident on the computing device and wherein the interface module receives program instructions from the home location register application in the application layer written for the second type of operating system and processes the instructions to select either one of the first number of APIs or one of the second number of APIs; and

a connection for connecting the computing device to a publicly switched telephone network (PSTN).

11. The system architecture of claim 10, wherein the interface module has a number of modules to translate instructions between the operating system layer and the application layer.

14. The system architecture of claim 10, wherein interface layer includes an operating system emulation module that includes translation and interpretation information therein.

15. The system architecture of claim 10, wherein the system architecture further includes an operating system emulation module to direct an instruction from the home location register application to an application program interface.

16. The system architecture of claim 10, wherein the system architecture further includes a number of component modules that can interface between an application designed for a second type of operating system and the operating system layer having a first type of operating system.

17. A method of executing an application comprising:

providing an application via an application layer having executable instructions to provide one or more applications to an operating system layer having executable instructions to provide a first type of operating system and

associated application program interfaces (APIs), wherein the associated APIs include a first number of APIs for operating on the first type of operating system that is resident on the computing device;

communicating instructions stored in memory and executable on a processor from the application to an interface module, wherein the interface module includes:

a second number of APIs for operating the home location register application with a second type of operating system that is not the type of operating system resident on the computing device and wherein the interface module receives program instructions from the application in the application layer written for the second type of operating system and processes the instructions to select either, one of the first number of APIs or one of the second number of APIs.

18. The method of claim 17, wherein processing the instructions from the application with the interface module includes using a list of instructions to be processed.

19. The method of claim 17, wherein the application is configured for a Linux based operating system.

20. The method of claim 17, wherein the application is configured for a Windows based operating system.

21. The method of claim 17, wherein the application is configured for a UNIX based operating system.

22. The method of claim 17, wherein the method further includes identifying instructions to be translated by the interface module.

23. A method of executing an application configured for a platform having first type of operating system on a platform having a second type of operating system comprising:

communicating instructions from the application to an interface module, the application configured for a first type of operating system;

interpreting the instructions from the application with the interface module by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on the second type of operating system and wherein the second number of APIs are resident on the interface module; and

communicating the instructions from the interface module to an operating system that is the second type of operating system.

24. The method of claim 23, wherein communicating instructions from the application to an interface module includes communicating instructions to an operating system emulation module within the interface module.

25. The method of claim 24, wherein interpreting the instructions includes directing an instruction from the operating system emulation module to an application program interface.

26. The method of claim 23, wherein communicating instructions from the application to an interface module includes communicating instructions to a network server emulation module within the interface module.

27. The method of claim 23, wherein interpreting the instructions includes translating an instruction configured for the first type of operating system to an instruction configured for the second type of operating system.

28. The method of claim 23, wherein interpreting the instructions includes converting a result configured for the second type of operating system to a result configured for the first type of operating system.

29. A computer readable medium having a set of computer executable instructions thereon for causing a device to perform a method, comprising:

communicating instructions from a telecommunications application to an interface module, the telecommunication application configured for a first type of operating system;

processing the instructions from the telecommunication application with the interface module by receiving program instructions from the application and processing the instructions to select either, one of a first number of APIs that are designed for use of the application on the first type of operating system and wherein the first number of APIs are resident on an operating system layer or one of a second number of APIs that are designed for use of the application on a second type of operating system and wherein the second number of APIs are resident on the interface module; and

communicating the instructions from the interface module to an operating system that is a second type of operating system.

30. The computer readable medium of claim 29, wherein communicating instructions from an application to an interface module includes communicating to an abstraction module within the interface module.

31. The computer readable medium of claim 29, wherein communicating instructions from an application to an interface module includes communicating instructions to a component module within the interface module.

32. The computer readable medium of claim 29, wherein the method further includes identifying instructions to be converted by the interface module.

## APPENDIX B

## Evidence Appendix

There is no extrinsic evidence to be considered in this Appeal. Therefore, no evidence is presented in this Appendix.

## APPENDIX C

## Related Proceedings Appendix

There are no related proceedings to be considered in this Appeal. Therefore, no such proceedings are identified in this Appendix.